

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-13741

(43) 公開日 平成7年(1995)1月17日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F	7/50	N		
	7/52	3 1 0 M		

審査請求 未請求 請求項の数 4 O L (全 9 頁)

(21) 出願番号	特願平5-150055
(22) 出願日	平成5年(1993)6月22日

(71) 出願人	000005821 松下電器産業株式会社 大阪府門真市大字門真1006番地
(72) 発明者	山下 智郎 大阪府門真市大字門真1006番地 松下電器産業株式会社内
(72) 発明者	若山 順彦 大阪府門真市大字門真1006番地 松下電器産業株式会社内
(72) 発明者	西村 明夫 大阪府門真市大字門真1006番地 松下電器産業株式会社内
(74) 代理人	弁理士 小鍛冶 明 (外2名)

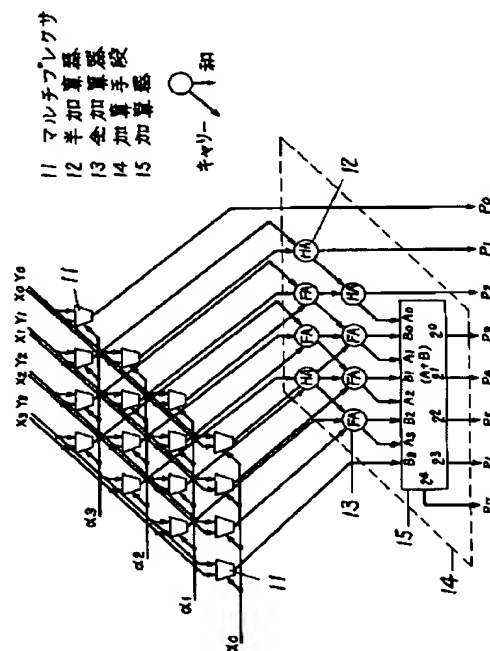
最終頁に続く

(54) 【発明の名称】 アルファ合成演算器

(57) 【要約】

【目的】 2つの数値 X、Y を  $\alpha$  の割合、 $P = \alpha X + (1 - \alpha) Y$  の式で合成するアルファ合成演算において、この合成演算を実現する回路の処理の高速化を目的とする。

【構成】 X、Y の各ビットをデータ入力、 $\alpha$  の各ビットを制御入力としてマルチプレクサ 11 に入力する。マルチプレクサ 11 の出力は、加算手段 14 に入力され、Wallace トリーにより総和が求められる。加算手段 14 はアルファ合成演算の結果 P を出力する。



1

## 【特許請求の範囲】

【請求項1】 2つの数値X、Yの各ビットをデータ入力、 $\alpha$ の各ビットを制御入力とする複数のマルチプレクサと、前記複数のマルチプレクサの出力からその総和を求める加算手段を備え、アルファ合成演算出力 $P = \alpha X + (1 - \alpha) Y$ を前記加算手段より得るアルファ合成演算器。

【請求項2】 複数のマルチプレクサの出力を、前記マルチプレクサのそれぞれの出力に数値Yを加算したものと

【請求項3】 複数のマルチプレクサの出力からその総和を求める加算手段に、Wallaceトリイを用いる請求項1記載のアルファ合成演算器。

【請求項4】 複数のマルチプレクサの出力からその総和を求める加算手段に、Wallaceトリイを用いる請求項2記載のアルファ合成演算器。

## 【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明はデジタル数値間の演算を行なうアルファ合成演算器に関する。

【0002】

【従来の技術】 近年、アルファ合成処理は、画像のデジタル数値化が進むにつれ、画像を扱う特殊効果、画像処理、画像生成等の装置において多用されている。アルファ合成処理は2種の画像をある比率で混合する処理であり、(数1)を全ての画素に対して適用することによって実現される。

【0003】

【数1】

$$P = \alpha X + (1 - \alpha) Y$$

30

\*

$$\begin{aligned} P &= \sum_{i=0}^3 \sum_{j=0}^3 (\alpha_i x_j + \overline{\alpha_i} y_j) 2^{i+j-4} \\ &= \sum_{i=0}^3 \sum_{j=0}^3 (\alpha_i x_j) 2^{i+j-4} + \sum_{i=0}^3 \sum_{j=0}^3 (\overline{\alpha_i} y_j) 2^{i+j-4} \\ &= \left( \sum_{i=0}^3 \alpha_i 2^i \right) \left( \sum_{j=0}^3 x_j 2^{j-4} \right) + \left( \sum_{i=0}^3 \overline{\alpha_i} 2^i \right) \left( \sum_{j=0}^3 y_j 2^{j-4} \right) \\ &= \alpha X + \overline{\alpha} Y \end{aligned}$$

【0009】 (数3)は、(数2)の右辺第3項が抜けた式であり、(数2)の近似式になる。 $\alpha$ のビット数が大きくなると近似は正確になる。図5に(数2)を完全に実現したアルファ合成演算器を示す。

【0010】 図4と同様X0～X3はXの最下位ビットから最上位ビット、Y0～Y3はYの最下位ビットから最上位ビット、 $\alpha 0 \sim \alpha 3$ は小数点を $\alpha 4$ にもつ4ビットの固定小数点 $\alpha$ の最下位ビットから最上位ビット、P

2

\* 【0004】 従って、1枚の合成画像を生成するには多数のアルファ合成演算を行なう必要があり、アルファ合成演算器は高速かつ小規模に実現されなければならない。従来のアルファ合成演算器には、特願平4-230902号に記載されたものがある。以下図面を参照しながら、前記した従来のアルファ合成演算器の一例について説明する。

【0005】 図4において、X0～X3はXの最下位ビットから最上位ビット、Y0～Y3はYの最下位ビットから最上位ビット、 $\alpha 0 \sim \alpha 3$ は小数点を $\alpha 4$ にもつ4ビットの固定小数点 $\alpha$ の最下位ビットから最上位ビット、P0～P7は8ビットの固定小数点数Pの最下位ビットから最上位ビットを示す。41はX、Yの各ビットをデータ入力とし、 $\alpha$ の各ビットを制御入力とするマルチプレクサである。42は半加算器(HA)、43は全加算器(FA)、44はマルチプレクサ41の出力を入力としPを出力する加算手段、45は加算器で、以上のように構成されたアルファ合成演算器について、以下その処理について説明する。図4の $\alpha$ の場合、(数1)は変形されて(数2)で表される。

【0006】

【数2】

$$P = \alpha X + \overline{\alpha} Y + Y 2^{-4}$$

【0007】 図4の処理を数式で表すと(数3)となる。

【0008】

【数3】

0～P7は、P3とP4の間に小数点を持つ8ビットの固定小数点数Pの最下位ビットから最上位ビットを示す。51はX、Yの各ビットをデータ入力、 $\alpha$ の各ビットを制御入力とするマルチプレクサである。52は半加算器、53は全加算器、54は加算器で、55はマルチプレクサ51の出力を入力としPを出力する、半加算器52、全加算器53、加算器54で構成された加算手段である。

3

【0011】処理において、図4と違うのは、(数2)を完全実現するために、Yを図4の処理結果に加算している点である。

【0012】(数1)通りにアルファ合成演算器を構成すると、1個の補数器、2個の乗算器、1個の加算器となり回路規模が大きくなるが、図4、図5の構成では乗算器1個程度の回路規模で済む利点がある。

【0013】

【発明が解決しようとする課題】しかしながら前記のような従来の構成では、マルチプレクサの出力からその総和を求める加算手段において反復セル型配列を使用しているので高速演算ができないという問題点を有していた。

【0014】本発明は前記問題点に留意し、小さい回路規模で高速のアルファ合成演算器を実現することを目的とする。

【0015】

【課題を解決するための手段】前記目的を達成するために本発明のアルファ合成演算器は、複数のマルチプレクサの出力からその総和を求める加算手段において、Wallaceトリートリーを用いるものである。

【0016】

【作用】前記の構成のアルファ合成演算器について、以下その動作について説明する。図2は、X、Y、 $\alpha$ が8ビット、Pが16ビットの近似値の場合のアルファ合成演算器を示している。図2において、21はマルチプレクサの出力、22は半加算器、23は全加算器、24は加算器を示す。25は、最上位ビットをP15、最下位ビットをP0、小数点をP7とP8の間とする出力結果Pを示す。

【0017】26はYの1ビットとXの積を、Yのビット番号だけ最上位ビット側にシフトしたものであり、右下の番号がシフト回数を示す。27は3段の26を入力とする加算処理部で、28、29は、それぞれ27の和の段とキャリーの段を示す。

【0018】図2のレベル1において、3段の26ごとに、それぞれ桁ごとの和とキャリーを求める。3段が和とキャリーの2段になるので、 $2 \times 2 = 4$ 段に未処理の2段を加えて6段に減る。

【0019】図2のレベル2において、レベル1での処理で減った6段を、再度、3段のグループに分けて和とキャリーを求める。 $6 \div 3 = 2$ グループでき、 $2 \times 2 = 4$ 段となる。同様に、レベルn-1でX段になったとすると、レベルnでの段数Yは、

【0020】

【数4】

$$Y = (X/3) + (X\%3)$$

【0021】となる。(数4)で「%」は余りを求める演算、「/」は商を求める演算を示す。以下、最終的に2

4

段になるまで同様の処理を続けるとレベル4で終了する。処理時間については、反復セル型では、 $8-1=7$ レベル、Wallaceトリートリーでは、4レベルであるため処理時間が大幅に短くなる。回路規模については、反復セル型の場合、最初の1段に半加算器が $8-1=7$ 個、残りの段はすべて全加算器となり、 $7 \times (7-1) = 42$ 個、Wallaceトリートリーの場合は、半加算器が15個、全加算器が38個となり、少々大きくなる程度である。

10 【0022】総和の対象段数がm段の処理時間について、反復セル型とWallaceトリートリーについて比較する。反復セル型では、段数分のレベルが必要であるため、m段の処理時間はO(m)となる。一方、Wallaceトリートリーでは、レベル数をLとすると(数5)よりおおよそその値が求まる。

【0023】

【数5】

$$L = 5.67(\log m - 0.30)$$

20 【0024】よってWallaceトリートリーでの処理時間は、O(log m)となる。前記2つの処理時間を比較すると、Wallaceトリートリーの方が高速であることがわかり、段数が多くなるほど処理時間の差は大きくなる。加算器24に桁上げ先見加算器を用いると、さらに処理が高速となる。

【0025】

【実施例】

30 【実施例1】以下に本発明の一実施例について、図面を参照しながら説明する。図1は本発明を最上位ビットをX3、最下位ビットをX0とする4ビットの整数Xと最上位ビットをY3、最下位ビットをY0とする4ビットの整数Yとを、最上位ビットを $\alpha$ 3、最下位ビットを $\alpha$ 0、小数点を $\alpha$ 4の上位とする4ビットの固定小数点数 $\alpha$ の割合で合成し、最上位ビットをP7、最下位ビットをP0、小数点をP3とP4の間とする8ビットの固定小数点数Pを出力とするアルファ合成演算器に適用した場合の一実施例である。

40 【0026】マルチプレクサ11は、X、Yの各ビットをデータ入力、 $\alpha$ の各ビットを制御入力とし、X、Yのビット長と $\alpha$ のビット長の積の数、すなわち16個存在する。配列状に接続された半加算器12と全加算器13で構成される加算手段14は、マルチプレクサ11の出力に接続され、加算器15は、桁上げが行なわれる最終段の加算処理をする。この場合、(数2)右辺第3項を無視しているため、Pは近似値である。

【0027】加算器15にたどり着くまでに要するレベル数は2回であり、同様の処理を反復セル型で行なうと3回であることと比較すると、処理時間が短縮されている。回路規模については、半加算器3個、全加算器5個で、反復セル型では、半加算器3個、全加算器5個とな

5

り回路規模も少し小さくなっている。

【0028】（実施例2）以下に本発明の第2の実施例について、図3を参照しながら説明する。図3において、31はマルチプレクサの出力、32は半加算器、33は全加算器、34は加算器を示す。35は、最上位ビットをP15、最下位ビットをP0、小数点をP7とP8の間とする出力結果Pを示す。36は、Yの1ビットとXの積を、Yのビット番号だけ最上位ビット側にシフトしたもので、右下の番号がシフト回数を示す。

【0029】37は3段の36を入力とする加算処理部10で、38、39は、それぞれ37の和の段とキャリーの段を示す。

【0030】図3では、Yを加算しているためX、Y、 $\alpha$ がいずれも8ビットの場合の完全なアルファ合成の実現となっている。

【0031】加算器34にたどり着くまでに要するレベル数は4回であり、同様の処理を反復セル型で行なうと、7回である。（実施例1）と比べて、処理時間の短縮効率が大きくなっている。回路規模については、半加算器が19個、全加算器が34個で、反復セル型では半加算器が7個、全加算器が39個となり、回路規模は、多少大きくなる程度である。

【0032】なお、実施例においては、X、Y、 $\alpha$ をすべて4ビットと8ビットのデジタル数値としたが、それ以外のビットの場合も、本発明のアルファ合成演算器を構成することが可能である。

6

【0033】さらに、X、Yのビット幅と $\alpha$ のビット幅が等しくない場合（例えばX、Yが8ビットで合成の割合 $\alpha$ が3ビット）でも、本発明を適用することが可能である。

【0034】

【発明の効果】以上のように本発明のアルファ合成演算器は、従来のアルファ合成演算回路の回路規模で、演算がmビットでは $O(\log m)$ の高速演算処理が実現される。

【図面の簡単な説明】

【図1】本発明の第1の実施例におけるアルファ合成演算器の構成を示すブロック図

【図2】本発明のアルファ合成演算器の処理過程を示す図

【図3】本発明の第2の実施例におけるアルファ合成演算器の構成を示すブロック図

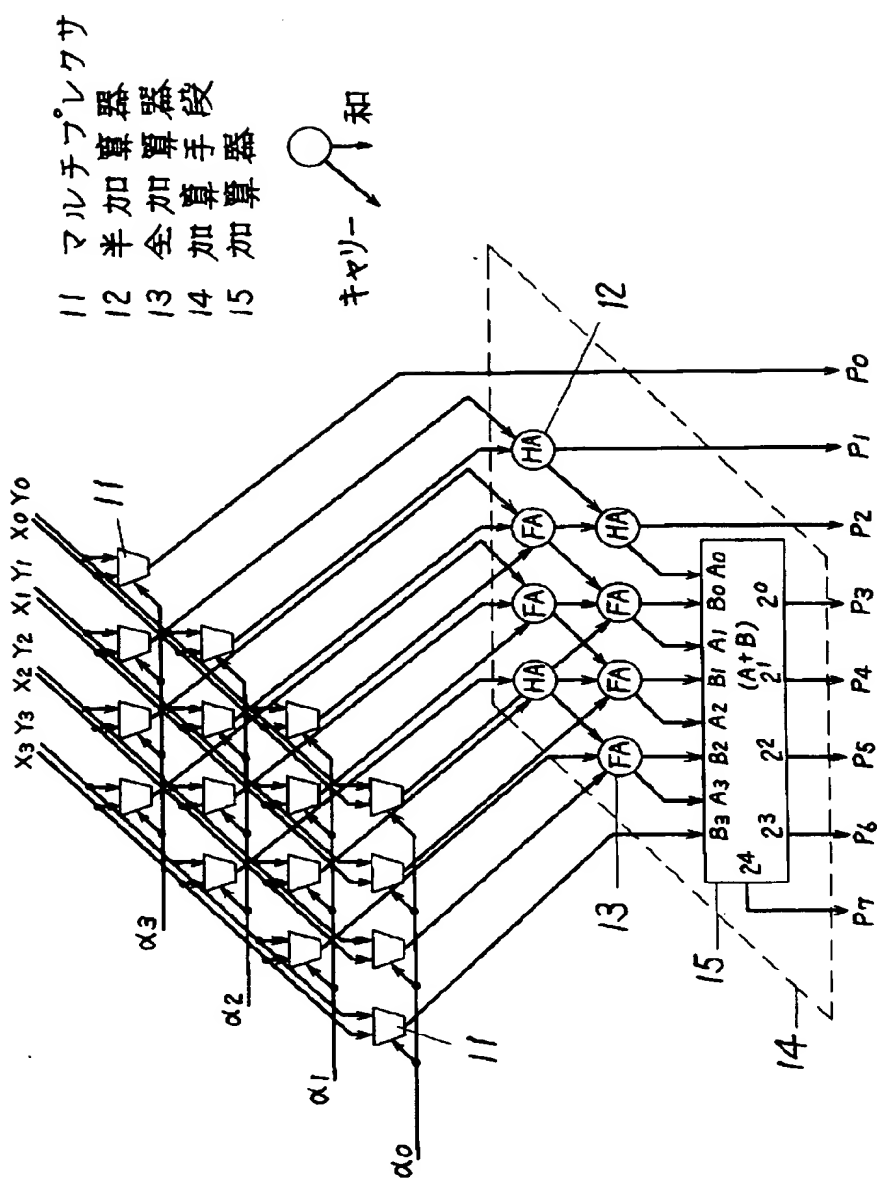
【図4】従来の近似値のアルファ合成演算器の構成を示すブロック図

【図5】従来のアルファ合成演算器の構成を示すブロック図

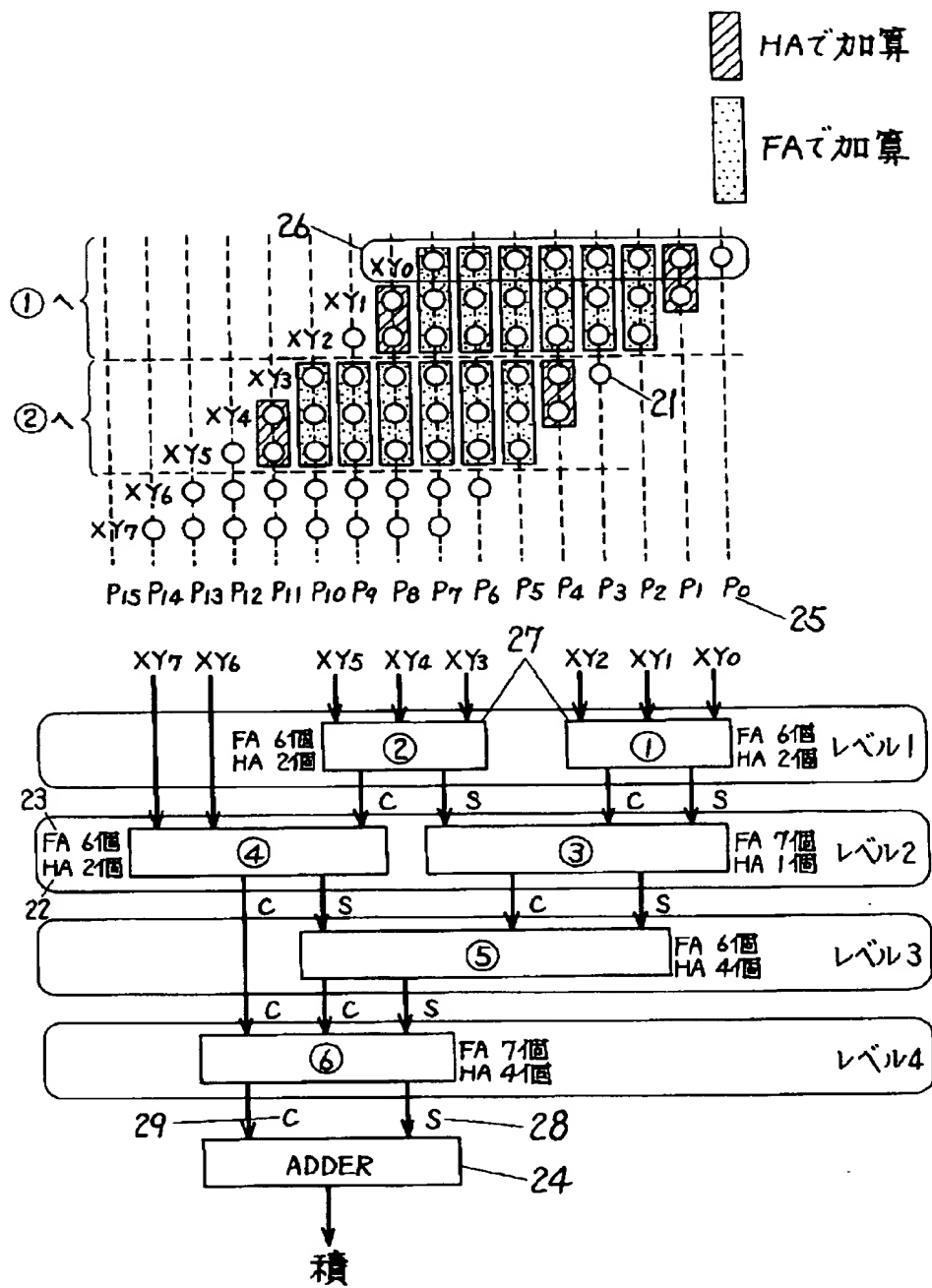
【符号の説明】

- 11 マルチプレクサ
- 12 半加算器
- 13 全加算器
- 14 加算手段
- 15 加算器

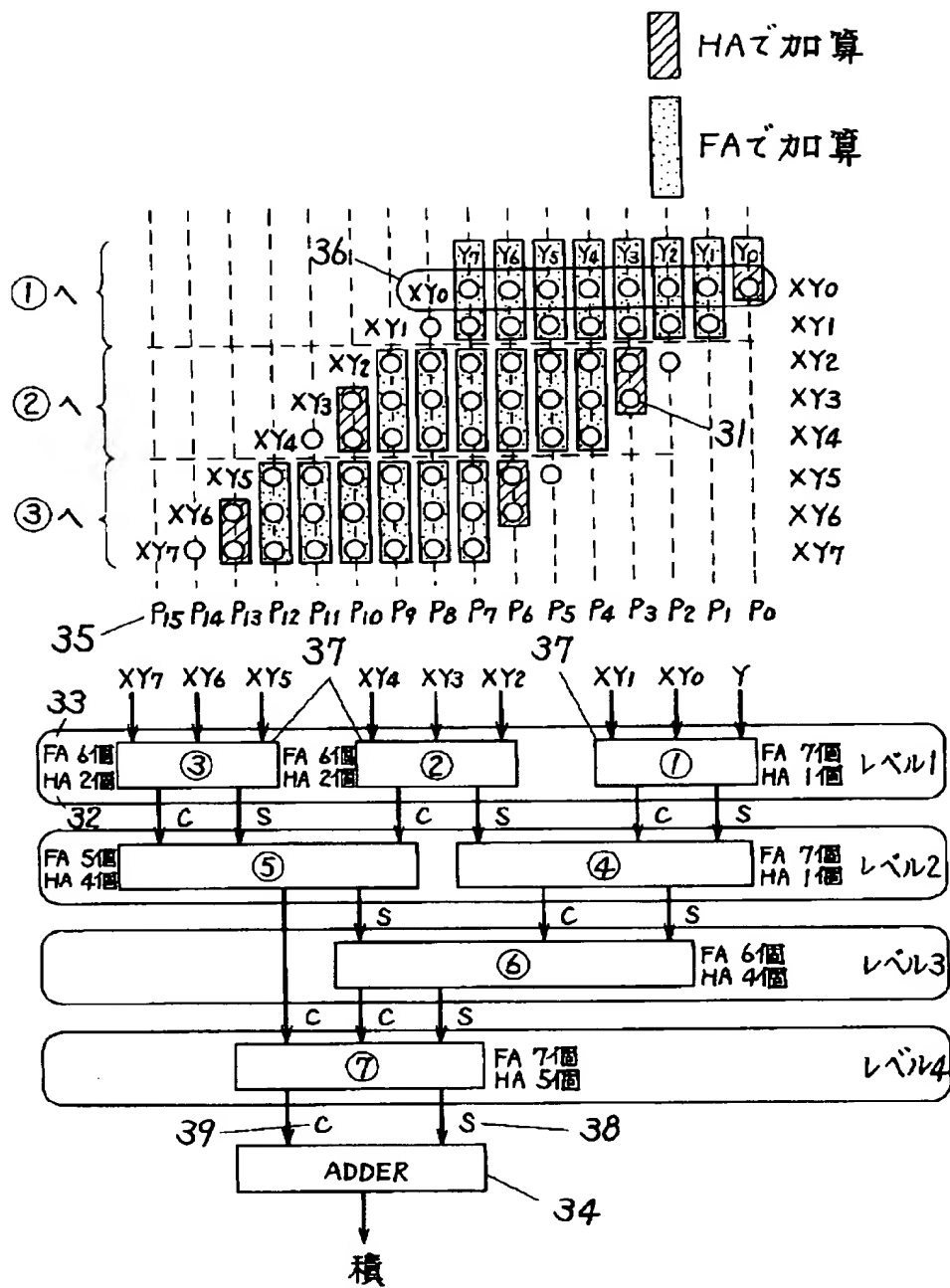
【図1】



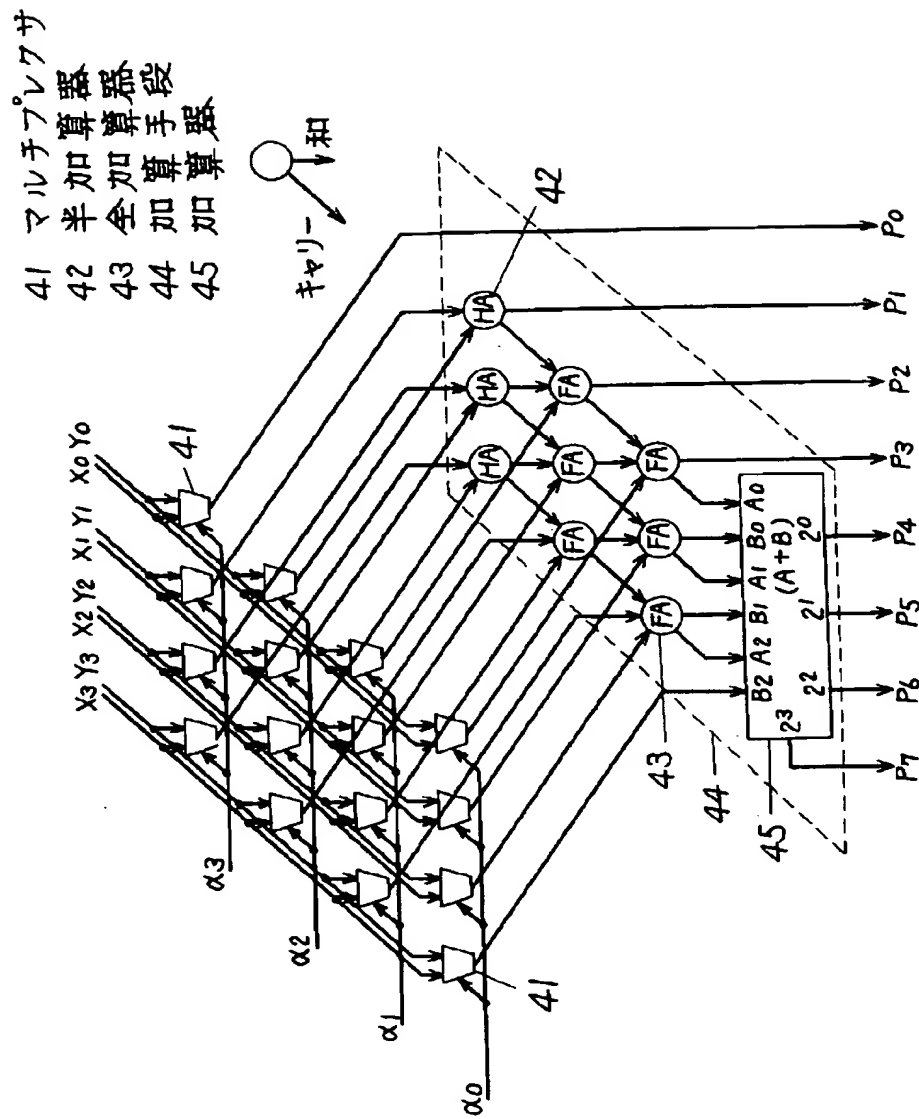
【図2】



【図3】

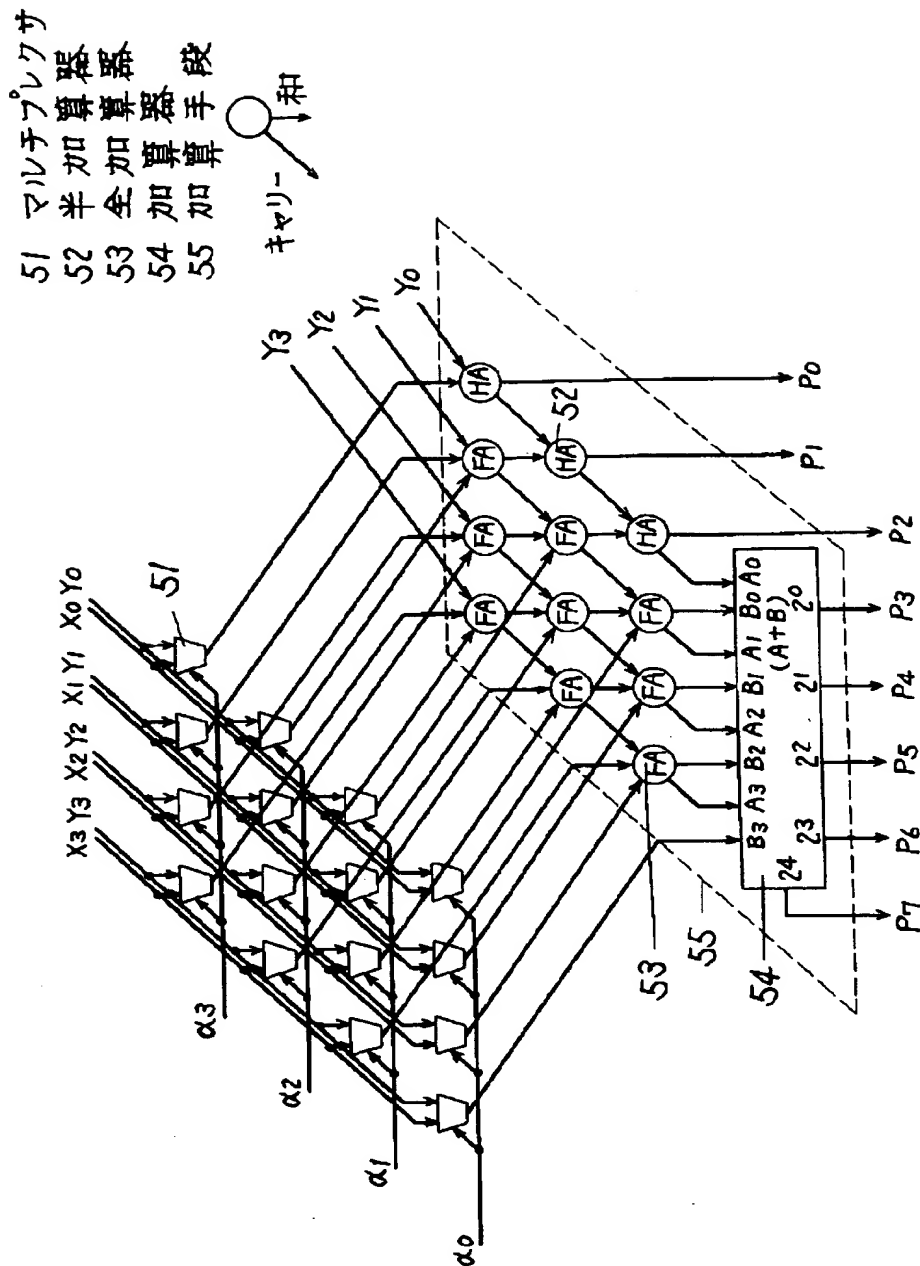


【図4】





【図5】



フロントページの続き

(72)発明者 西澤 貞次  
大阪府門真市大字門真1006番地 松下電器  
産業株式会社内